Data-driven reinforcement learning-based parametrization of a thermal model in induction traction motors

Anas Fattouh[a]*, Smruti Sahoo[b]

[a] *School of Innovation, Design and Technology (IDT), Mäalardalen University, Eskilstuna, Sweden,*

[b] *Alstom, Västerås, Sweden*

*Corresponding author: anas.fattouh@mdu.se

Abstract

Monitoring the temperature of induction traction motors is crucial for the safe and efficient operation of railway propulsion systems. Several thermal models were developed to capture the thermal behaviour of the induction motors. With proper calibrating of the thermal model parameters, they can be used to predict the motor's temperature. Moreover, calibrated thermal models can be used in simulation to evaluate the motor's performance under different operating conditions and find the optimal control strategies.

Parameterization of the thermal model is usually performed in dedicated labs where the induction motor is operated under predefined operating conditions and calibrating algorithms are then used to find the model's parameters. With the development of digital tools, including smart sensors, Internet of Things (IoT) devices, software applications, and various data collection platforms, operational data can be collected and used later to calibrate the parameters of the thermal model. Nevertheless, calibrating the model's parameters from operational data collected from different driving cycles is challenging as the model has to capture the thermal behaviour from all driving cycles' data.

In this paper, a data-driven reinforcement learning-based parametrization method is proposed to calibrate a thermal model in induction traction motors. First, the thermal behaviour of the induction motor is modelled as a thermal equivalent network. Second, a reinforcement learning (RL) agent is designed and trained to calibrate the model parameters using the data collected from multiple driving cycles. The proposed method is validated by numerical simulation results. The results showed that the trained RL agent came up with a policy that adeptly handles diverse driving cycles with different performance characteristics.

Keywords: Railway, Propulsion system, Traction motor, Induction motor, Thermal model, Parametrization, Data-driven, Reinforcement learning, Calibrating, Optimization

1. Introduction

Traction motors are subjected to varying operating and environmental conditions due to the dynamic loads over the operation cycle. The transient loads may cause overloading of the drive components which causes extra heat load. Operations causing overheating of the motor parts are of significant concern as they may lead to stator winding failure and accelerated ageing. Furthermore, to be able to exploit the motor's maximum utilization, it is essential that its operation is optimized to make it cost-effective.

On the other hand, induction motors (IMs) are the most used motors in railway propulsion applications to date because of their mechanical robustness and high overload capabilities. The added advantages are their low cost and the possibility of employing multiple drives connected to a single converter (Nategh *et al.*, 2020). However, their performance varies nonlinearly with temperature, frequency, saturation, and operating point which makes temperature monitoring essential for the safe and reliable operation of the motor.

The thermal limits of these motors are associated with the winding insulation material which is classified based on its temperature withstanding capacity. There are several established direct or indirect means for estimating the temperature in motor parts. Direct methods such as installing contact-based sensors in the stator, and rotor are the simplest means for measurement. However, the data transmission in the rotating parts has to be carried out with the help of end slip rings, or telemetry means. Regardless, installing sensors requires integration effort and additional cost and adds complexity due to their inaccessibility for replacement in case of failures or detuning. Hence model-based measurement techniques have been rather focused in the past decade (Ramakrishnan *et al.*, 2009; Wilson, 2010). Here the temperatures can be estimated from the temperature dependent electrical parameters both off-line and online manner. In these approaches, it is imperative that the model dynamic behaviours are accurately accounted for to avoid any estimation errors. These are also invasive in nature and create disturbance to the normal operation.

Computational fluid dynamics (CFD) and heat equation-based finite element analysis (FEA) are powerful techniques for accurate temperature estimation. However, due to their rigorous modelling effort and high computational power and time, they have been excluded from real-time monitoring upfront (Kirchgässner *et al.*, 2021).

An alternative and computationally light temperature estimation technique is using the lumped-parameter thermal network (LPTN) model. An LPTN model summarizes the heat transfer process and can be represented in thermal equivalent circuit diagrams with knowledge of the used material and geometry and based on heat transfer theory. In this model, the thermal parameters are thermal resistances and capacitance values (Chen *et al.*, 2020). The LPTNs can be designed to estimate the temperatures at several or fewer locations in the motor. However, the LPTN model needs accurate distributed motor losses, surface contact thermal conductance and heat transfer convection coefficients information, hence, needs to be calibrated empirically (Huber *et al.*, 2014).

The complexity of LPTN models depends on the number of chosen nodes in the network. Generally, white box LPTN, based on pure analytical equations, are more accurate but are endowed with many thermal parameters, which could be complex to calculate in practice. A low-order structure with fewer nodes is computationally lightweight. In the past, several reduced order models categorized as light grey (5-15 nodes), and dark grey (2-5 nodes) LPTN models were structured to estimate the temperature with good accuracy. In this approach, only the dominant heat transfer pathways are represented, hence, expert domain knowledge is essential for the correct choice of not only their parameter values but also for their structural design (Wallscheid and Böcker, 2015). These models have proven to have good estimation performance, provided the parameters are identified appropriately. The use of a reduced order LPTN method would require estimations of many parameters that are not well known or possible to calculate using analytical equations. Thus, the identification of the parameters is an important step in these studies.

To date, several research works on the parameterization of the LPTN model have been performed using empirical measurements by applying different methods (Guemo *et al.*, 2013; Huber *et al.*, 2014; Xiao and Griffo, 2020). The proposed identification procedure varies the parameter values until the used LPTN model gives the same results as the experimental ones. As described by (Wallscheid, 2021), parameter identification can be made in a local approach or a global approach. In the work presented by (Huber *et al.*, 2014), a three-node LPTN is parametrized based on a global approach. A sequence of interdependent identification steps was followed, and the experimental data are used to find the thermal parameters. The model uses the measurement-based loss inputs available with motor electronic control unit (ECU) quantities, such as motor speed and electric currents. The parameter identification approach has been built on the idea of mapping the linear time-varying parameters to a set of time-invariant models operating within a certain chosen environment. Thus, a consistent parameter set for the whole operating region could be obtained with the adaptation of the relevant boundary conditions through various identification cycles. While the global approach is more robust in capturing all operating regions of the motor than the local approach, it can also be problematic if the parameter landscape to be identified is large and highly nonlinear in nature. Hence it is complex to find the parameter values near the global parameter optimum (Wallscheid, 2021). However, the global approach captures the nonlinearity in the form of a parametrizable function and, hence, they are potentially more versatile compared to the global and local approaches.

The use of inverse methods is also popular for finding thermal parameters. In the work presented by (Guemo *et al.*, 2013), the identification of the parameters is made by solving the optimization problem using determinist inverse-based methods such as the Gauss-Newton method, the Levenberg-Marquardt method, and stochastic inverse-based methods such as the Genetic Algorithms. These concept methods are used to minimize the residuals between measured and calculated temperatures. (Sciascera *et al.*, 2017) employed a tuning procedure based on a sequential quadratic programming iterative method for obtaining the uncertain thermal parameters of the thermal network. However, the computation cost of such tuning procedures is high due to the time-variant nature of the parameters. Furthermore, to improve computational efficiency, the dependence of the state matrix on the phase current is approximated with polynomial approximation.

The temperature rise in the electric motor occurs due to the electro-mechanical power conversion losses. Winding and core losses and mechanical and windage losses are the prominent ones for an induction motor. The winding losses can be calculated for a given winding resistance and measured current. However, the winding resistance changes based on the temperature which is a state variable in the thermal matrix. Furthermore, the core loss is not measurable. A usual approach to determining iron losses is measuring total power losses and subtracting winding losses. Hence, all errors in the determination of total and winding losses directly add up to an error in the iron losses values. To deal with these uncertainties, (Gedlu *et al.*, 2020) used an extended iron loss model as input to a low-order LPTN model for temperature estimation. The loss inputs as a form of spatial loss model calculate individual core losses for each node. In addition to the heat transfer coefficients, the uncertain parameters in the core loss equations are calibrated in their possible searching space using particle swarm optimization (PSO) to minimize the estimation error in comparison to empirical measurements.

Xiao and Griffo (Xiao and Griffo, 2020) in their work presented an online measurement-informed thermal parameter estimation using a recursive Kalman filter method. While a Pulse-Width Modulation (PWM)-based estimation method is utilized for rotor temperature measurement, the temperatures in three nodes such as stator core, winding, and rotor are predicted. The input losses for the LPTN model are derived based on a model-based approach and with the use of Finite Elements (FE) analysis. The identification problem is formulated as a state observer with eight states. Three of the states correspond to the nodes' temperatures and the rest five states represent the unknown thermal resistances parameters in the LPT network. The non-linearity of the model is dealt with continuous updated linearization the extended Kalman filter method.

The growing interest and upsurge in machine learning (ML) techniques in the past decade make these potentially viable tools in the area of automated monitoring and motor drive control. A pure ML model, i.e. a model without expert knowledge of any classic fundamental heat theory, can be trained to estimate the temperature empirically. In this case, the model parameters are fitted based on collected testbench/observation data only (Kirchgässner *et al.*, 2021; Wallscheid, 2017). The widely used ML algorithm is the linear regression technique which has low computational complexity and is used for temperature predictions (Kirchgässner *et al.*, 2019; Zhu, 2019). However, as linear regression is a linear time-invariant, it does not capture the dynamics of the motor model.

In the field of sequence learning tasks and in high dynamics, recurrent neural networks and convolutional neural networks are the state of the art in classification and estimation performance. In the study conducted by (Kirchgässner *et al.*, 2019), deep recurrent and convolutional neural networks with residual connections are empirically evaluated for predicting temperature profile in the stator teeth, winding, and yoke as well. The concept is to parameterize neural networks entirely on empirical data, without being driven by the domain expertise. Furthermore, supervised ML models are also investigated for online parameter estimation such as rotor resistance and mutual inductance in the control system of an induction motor (Wlas *et al.*, 2008). In the presented work, a simple two-layer artificial neural network (ANN), consisting of an input layer, one hidden layer, and an output layer, is trained by minimizing the error between the rotor flux linkages based on an induction motor analytical voltage model and the output of the ANN-trained model. Feedforward and recurrent networks are used to develop an ANN as a memory for remembering the estimated parameters and for computing the electrical parameters during the transient state.

While pure data-driven ML models[1] are effective in predicting the temperature, their parameters are not interpretable and could not be designed with the low amount of model parameters as in an LPTN model at equal estimation accuracy. As a further development, to make expert knowledge-based calibration less desirable and to account for the uncertainties regarding the input power losses, (Kirchgässner W. W., 2023) proposed a deep learning-based temperature model where a thermal neural network is introduced, which unifies both consolidated knowledge in the form of heat-transfer-based LPTNs, and data-driven nonlinear function approximation with supervised machine learning.

The reinforcement learning (RL)-based methods are other promising data-driven techniques explored in the field of control of electric motor drives (Book *et al.*, 2021). RL methods enable learning in a trial-and-error manner and avoid supervision of each data sample. The algorithm requires a reward function to receive the reward signals throughout the learning process. Thus, the control policy could be improved on a continuous basis based on the measurement feedback (Sutton and Barto, 2018).

In this paper, a data-driven reinforcement learning-based parametrization method is proposed to calibrate a thermal model of an induction traction motor.

The rest of the paper is organized as follows. Section 2 presents the parametrization of an LPTN Model for an induction motor. Section 3 explains the developed RL framework to calibrate the parameters of the parametrized LPTN model. The dataset and the training process are given in Section 4. Section 5 shows and discusses the results of the calibrated thermal model. Concluding remarks and future work are given in Section 6.

[1] Models without expert knowledge.

2. Parametrizing the Thermal Model

From a thermal point of view, the motor is modelled with four nodes: stator winding (node 1), stator core (node 2), rotor winding (node 3) and rotor core (node 4). The thermal equivalent network is illustrated in Figure 1 with thermal capacitances, to which a power (heat) source is connected, and with thermal conductance among the nodes and to the cooling air.

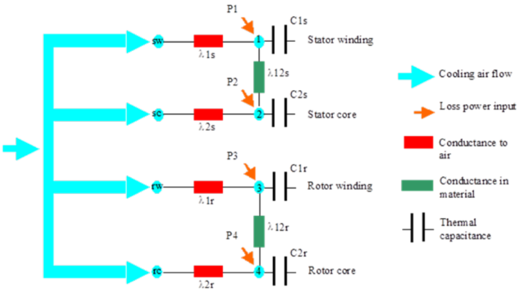


Figure 1: Lumped Parameter Thermal Network Model

Thermal capacitances C_{1s}, C_{2s}, C_{1r}, and C_{2r} values are calculated analytically from the geometry and material information of the motor. The capacitance for stator yoke C_{1s} is the sum of the capacitance of stator housing, stator back iron, stator tooth and flange mounted. The stator winding capacitance C_{2s} includes the capacitance for the stator winding and the end winding capacitances. The capacitance for stator yoke $C_{1\mathrm{r}}$ is the sum of the capacitance of rotor yoke, and rotor bars. The rotor winding capacitance C_{2r} includes the capacitance for the rotor winding and the end winding capacitances.

The thermal conductance λ_{1s}, λ_{2s}, λ_{1r}, λ_{2r} vary with the airflow due to the convection. The model shown in Figure 1 can be represented mathematically by the following first-order differential system:

$$P_1 = C_{1s}\frac{dT_1}{dt} + \lambda_{1s}(T_1 - T_{env}) + \lambda_{12s}(T_1 - T_2) \quad (1)$$

$$P_2 = C_{2s}\frac{dT_2}{dt} + \lambda_{2s}(T_2 - T_{env}) + \lambda_{12s}(T_2 - T_1) \quad (2)$$

$$P_3 = C_{1r}\frac{dT_3}{dt} + \lambda_{1r}(T_3 - T_{env}) + \lambda_{12r}(T_3 - T_4) \quad (3)$$

$$P_4 = C_{2r}\frac{dT_4}{dt} + \lambda_{2r}(T_4 - T_{env}) + \lambda_{12r}(T_4 - T_3) \quad (4)$$

where T_i is the temperature at the corresponding node i. The temperatures of the cooling air at the four nodes (marked as sw, sc, rw and rc in Figure 1) are assigned to the environment (or ambient) temperature T_{env}.

The losses at the four nodes in Figure 1 are distributed as shown in Table 1 and they can be calculated as follows:

$$P_1 = K_{stemp}P_{cu1} + K_{stray}P_{stray} + K_{harm}P_{harm} \quad (5)$$

$$P_2 = K_{Pfe}P_{fe} \quad (6)$$

$$P_3 = K_{rtemp}P_{cu2} + (1 - K_{stray})P_{stray} + (1 - K_{harm})P_{harm} \quad (7)$$

$$P_4 = (1 - K_{Pfe})\ P_{fe} \quad (8)$$

where $P_{cu1}, P_{cu2}, P_{stray}, P_{harm}, P_{fe}$ are the stator copper loss, rotor copper loss, stray loss, harmonic loss, and iron loss respectively. The coefficients K_{stemp}, K_{rtemp}, K_{stray}, K_{harm} and K_{Pfe} are the corresponding losses coefficients.

Table 1: Loss Distribution in the LPTN Model.

Node	Winding Losses	Stray Losses	Harmonic Losses	Iron Losses
1	x	x	x	
2				x
3	x	x	x	
4				x

The losses in Equations (5)-(8) can be calculated as follows (Kral *et al.*, 2013; Filizadeh, 2013; Maroteaux, 2016; Nasir, 2020; IEC/TS, 2010):

$$P_{cu1} = R_1\ I_1^2 \quad (9)$$

$$P_{cu2} = R_{21}\ I_{21}^2 \quad (10)$$

$$P_{stray} = P_{SUP}\left(\frac{f}{f_{nom}}\right)^{1.5}\left(\frac{I_1}{I_{1,nom}}\right)^2 \quad (11)$$

$$P_{fe} = K_f\ f^{\alpha}B_{max}^{\beta} \quad (12)$$

where I_1, I_{21} are the stator and rotor currents respectively, R_1 and R_2 are the stator and rotor winding resistances respectively which depend on the temperature according to following equations:

$$R_1 = R_{1,20} * (1 + \alpha_{R1} * (T_1 - 20) \quad (13)$$

$$R_{21} = R_{21,20} * (1 + \alpha_{R2} * (T_3 - 20) \quad (14)$$

where $R_{1,20}$, $R_{21,20}$ are the stator and rotor winding resistance at 20 °C and α_{R1}, α_{R21} the temperature coefficient of stator and rotor respectively. In Equation (11), f is the stator frequency with a nominal value f_{nom}, I_1 is the stator current with a nominal value $I_{1,nom}$ and P_{SUP} is equivalent rated input power. In Equation (12), K_f is a constant that depends on the material properties and the core geometry, f is the frequency of the magnetic field, B_{max} is the peak magnetic flux density in the core and α and β are empirically determined constants. The harmonic losses P_{harmo} is measured at few operation points and included as a look- up table in the loss model.

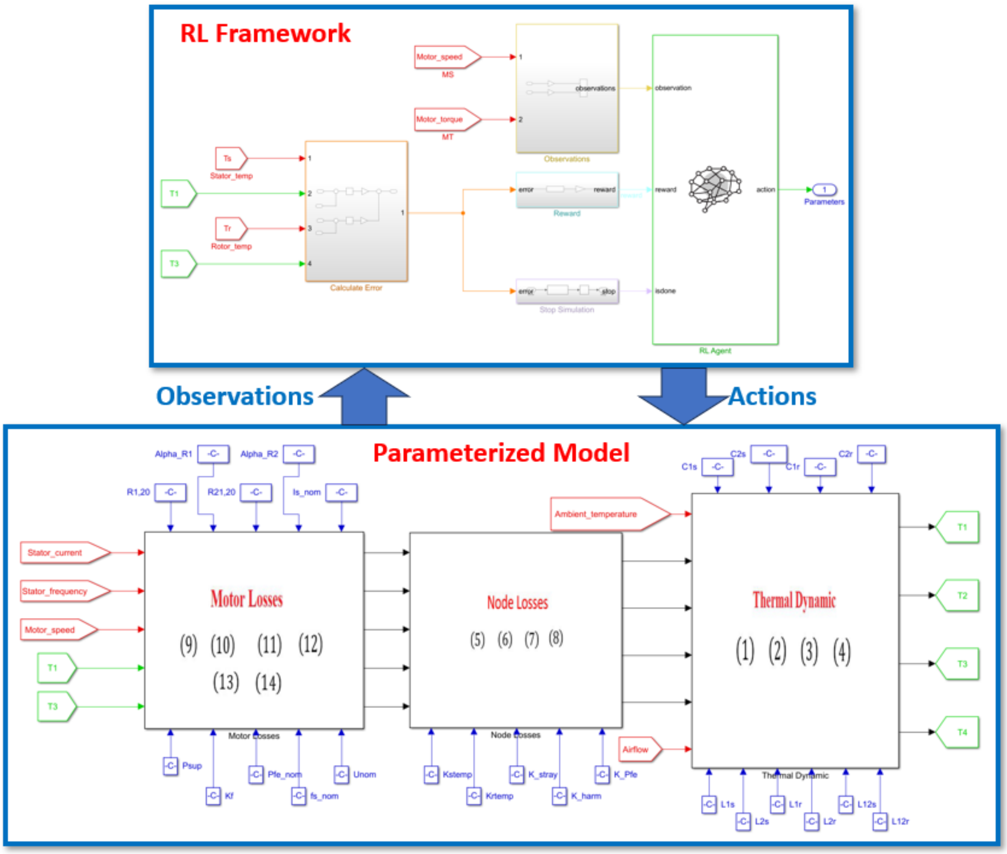


Figure 2: Reinforcement Learning Framework

3. The Reinforcement Learning Framework

The model developed in the previous subsection has been implemented as a parameterized model and employed within a reinforcement learning (RL) framework as shown in Figure 2.

3.1. The Parameterized Model

The parameterized model represents the dynamic evolution of the temperature in the induction motor according to the model developed in Section 2. The model inputs (the red signals) are the stator current, the stator frequency, the motor speed, the airflow of the cooling air and the ambient temperature. The model outputs (the green signals) are the temperatures (T_1, T_2, T_3, T_4) at the four nodes (P_1, P_2, P_3, P_4) respectively of the LPTN model shown in Figure 1. The blue signals in Figure 2 represent the parameters of Equations (1)-(14) explained in Section 2.

3.2. The RL Framework

The RL framework consists of the RL agent, the observations, the reward function and the actions.

3.2.1. The Observations

The observations represent the information that the RL agent can sense from the environment (the parameterized model) during the operation of the induction motor. Some observations, such as the motor speed (MS) and the motor torque (MT), are used directly by the agent to sense any changes in the motor's operating conditions. Other observations, such as the measured stator and rotor temperatures (*Ts, Tr*) and the model outputs (T_1, T_3), are used to calculate the reward. It should be noted that T_2 and T_4 could be considered among the observations if there are related measurements. However, in this work, there are no measurements related to T_2 and T_4.

3.2.2. The Reward Function

The reward function produces a value that reflects the effectiveness of the agent's actions in the environment. This value serves as a critical signal guiding the agent's learning process towards achieving its goals effectively. The reward value encapsulates the objectives of the RL problem, which is, in our context, minimizing the error between the measured temperatures (*Ts, Tr*) and the model's output temperatures (T_1, T_3). Hence, the reward function is given by:

$$r = \frac{-\omega_1}{\omega_2|T_s - T_1| + \omega_3|T_r - T_3| + \omega_4} \qquad (15)$$

where ω_1, ω_2, ω_3 and ω_4 are positive weights.

3.2.3. The Actions

The actions represent all the possible values of the model parameters (the blue signals in the parameterized model). Actions are computed by the agent based on the observations and the reward value and using a policy that is optimized during the training process to maximize the expected cumulative reward over time. The policy is essentially the agent's strategy for selecting actions in different situations to achieve its objectives efficiently.

3.2.4. The RL Agent

The RL agent is composed of two main elements: a policy and a learning algorithm. The policy maps the observations with actions to be taken while the learning algorithm updates the policy parameters, based on the actions, observations and rewards, to maximize the expected cumulative long-term reward. During the learning process and depending on the learning algorithm, the agent retains two types of models: critic and actor models. The critic model predicts the expected cumulative reward (Q-Value) from a given observation and action that is later used by the actor model to return the action that maximizes the predicted discounted cumulative long-term reward (Sutton and Barto, 2018).

In this work, a TD3[2] algorithm is used in the RL framework shown in Figure 2. TD3 agent works in a continuous environment and has improved policy model performance over time (Dankwa *et al.*, 2019; Nicola and Nicola, 2021). Moreover, the episodic training paradigm enables the TD3 agent to select different training datasets after each episode. This will allow the agent to find the optimal policy (the thermal model parameters) from measurements recorded during different driving cycles, which is the main objective of this work.

In the following section, the dataset and the training process are explained.

4. Training the RL Agent

4.1. The Dataset

The dataset represents the data recorded from the induction motor during the operation of nine different driving cycles. It is composed of all the red signals shown in Figure 2, i.e., the motor speed, the motor torque, airflow, the stator current and frequency, the motor voltage, the stator winding temperature and the rotor winding temperature.

It should be noted that the data are recorded at different sampling frequencies with some missing values that require resampling the dataset and interpolating the missing values.

It should also be noted that the dataset is not used directly by the RL framework to train the agent, but it is used in the parameterized model which is considered an unknown environment to the agent.

4.2. The TD3 Structure

The TD3 algorithm employs one actor model (network) and two critic models (networks) as shown in Figure 3.

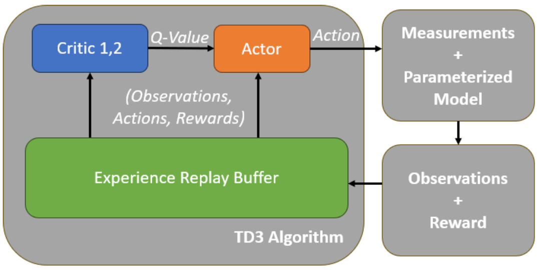


Figure 3: Simplified TD3 Structure

The actor and critic models are approximation models, such as a deep neural network, that are trained from *(Observations, Actions, Rewards)* sampled from the experience replay buffer.

4.3. The Training Process

The RL agent is trained following an episodic scheme. Episodes are used to model the concept of a task or problem that the agent is trying to learn (Sutton and Barto, 2018). An episode refers to a single run of the agent's interaction with the environment, starting from an initial state and continuing until a terminal state is reached.

Each episode consists of a specific number of discrete time steps. At each discrete time step:

1. The actor applies an action based on the current observation and expected Q-value.
2. The new observation, action and reward are stored in the experience replay buffer.
3. A random batch of experiences is sampled from the experience replay buffer and used to update the parameters of the critic models by minimizing a loss function across all sampled experiences.
4. After some specified steps, the parameters of the actor model are updated using a sampled policy gradient that maximizes the expected discounted reward.

5. Results and Discussion

5.1. Preprocessing the Dataset

In this work, nine driving cycles have been used to represent the unknown environment which is used to train the agent to find the optimal values of the thermal conductance λ_{1s}, λ_{2s}, λ_{1r} and λ_{2r}. As mentioned previously, data are recorded at different sampling frequencies with some missing values that

[2] Twin-Delayed Deep Deterministic Policy Gradient

require resampling the dataset and interpolating the missing values.

Figure 4 shows a sample of measured motor speed and motor torque from one driving cycle (DC1) after resampling and interpolating the missing values.

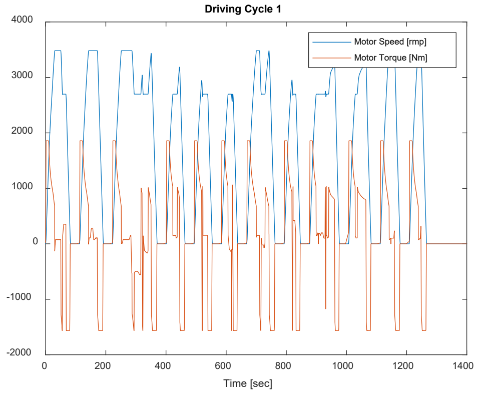
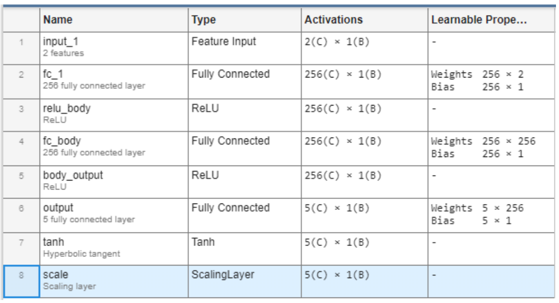


Figure 4: A Sample of Motor Speed and Torque (DC1)

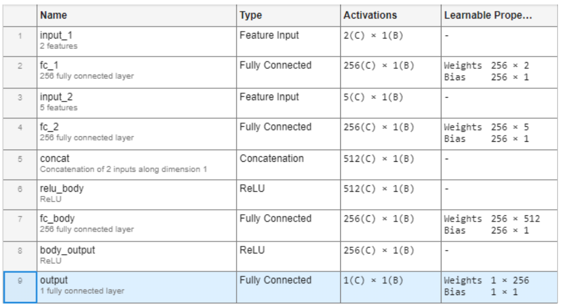
5.2. Training the RL Agent

The RL agent was trained using the actor and critic deep neural network (DNN) models with the layers depicted in Figures 5 and 6.



	Name	Type	Activations	Learnable Prope...
1	input_1 2 features	Feature Input	2(C) × 1(B)	-
2	fc_1 256 fully connected layer	Fully Connected	256(C) × 1(B)	Weights 256 × 2 Bias 256 × 1
3	relu_body ReLU	ReLU	256(C) × 1(B)	-
4	fc_body 256 fully connected layer	Fully Connected	256(C) × 1(B)	Weights 256 × 256 Bias 256 × 1
5	body_output ReLU	ReLU	256(C) × 1(B)	-
6	output 5 fully connected layer	Fully Connected	5(C) × 1(B)	Weights 5 × 256 Bias 5 × 1
7	tanh Hyperbolic tangent	Tanh	5(C) × 1(B)	-
8	scale Scaling layer	ScalingLayer	5(C) × 1(B)	-

Figure 5: Layers of the Actor DNN Model



	Name	Type	Activations	Learnable Prope...
1	input_1 2 features	Feature Input	2(C) × 1(B)	-
2	fc_1 256 fully connected layer	Fully Connected	256(C) × 1(B)	Weights 256 × 2 Bias 256 × 1
3	input_2 5 features	Feature Input	5(C) × 1(B)	-
4	fc_2 256 fully connected layer	Fully Connected	256(C) × 1(B)	Weights 256 × 5 Bias 256 × 1
5	concat Concatenation of 2 inputs along dimension 1	Concatenation	512(C) × 1(B)	-
6	relu_body ReLU	ReLU	512(C) × 1(B)	-
7	fc_body 256 fully connected layer	Fully Connected	256(C) × 1(B)	Weights 256 × 512 Bias 256 × 1
8	body_output ReLU	ReLU	256(C) × 1(B)	-
9	output 1 fully connected layer	Fully Connected	1(C) × 1(B)	Weights 1 × 256 Bias 1 × 1

Figure 6: Layers of the Critic 1 and 2 DNN Models

The training steps explained in Subsection 4.2 were applied with the parameters shown in Table 2.

Table 2: Training Parameters.

Property	Value
Max Episodes	1000
Max Steps per Episode	9990
Averaging Window Length	100
Stop Training Value	-190
Agent Sample Time	0.1

Figure 5 shows the training process where the blue curve represents the episode reward, the red curve represents the average reward and the orange curve represents the estimated cumulative rewards at the beginning of each training episode. The figure shows that the agent learned an optimal policy (parameters) after 101 episodes.

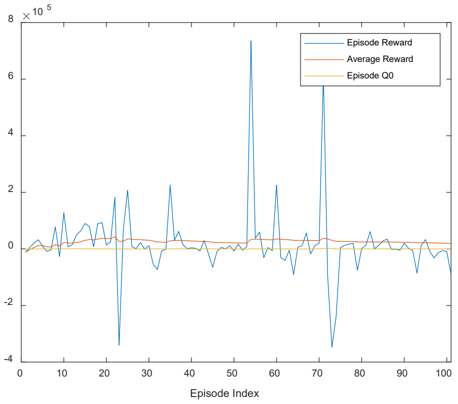


Figure 5: Training Process

5.3. Validating the Trained Agent

Validation of the RL agent usually involves periodically evaluating the learned policy directly in the environment.

When evaluating the learned policy with different driving cycles, the agent came up with the following parameters:

$\lambda_{1sn} = 1, \lambda_{1rn} = 40, \lambda_{2sn} = 20, \lambda_{2rn} = 0.0001, Q_n = 10^{-9}$

Figures 7 and 8 show the measured and model temperatures for driving cycle 1 (DC1) and driving cycle 8 (DC8) respectively.

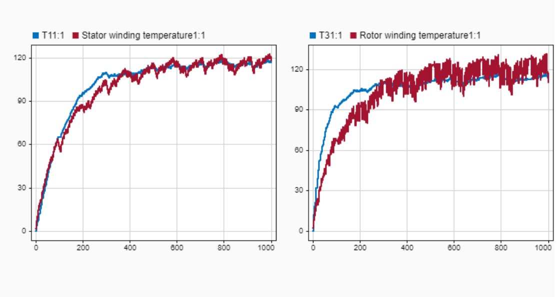


Figure 7: Model Parameters and Measured and Model Temperatures for DC1

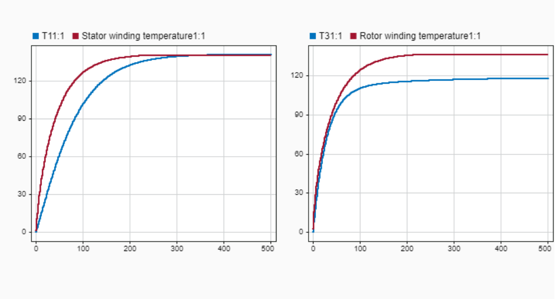


Figure 8: Model Parameters and Measured and Model Temperatures for DC8

6. Conclusion and Future Work

In the paper, a reinforcement learning framework is proposed for training an agent to find the parameters of the thermal model in induction traction motors. The framework has been applied to find the thermal conductance for the thermal network model from nine driving cycles.

By running different driving cycles, the trained agent came up with a policy that produces the parameters for the different driving cycles. The model with the calibrated parameters showed a good estimation of stator and rotor temperature.

In future work, other structures for the agent and the reward function will be considered to produce better temperature estimation.

Acknowledgment

This research work has received funding through the AIDOaRt project from the ECSEL Joint Undertaking (JU) under grant agreement No 101007350.

References

Book, G., Traue, A., Balakrishna, P., Brosch, A., Schenke, M., Hanke, S., Kirchgässner, W. and Wallscheid, O. (2021) 'Transferring online reinforcement learning for electric motor control from simulation to real-world experiments', *IEEE Open Journal of Power Electronics*, 2, pp. 187-201.

Chen, B., C. Wulff, K. Etzold, P. Manns, G. Birmes, J. Andert and S. Pischinger (2020) 'A comprehensive thermal model for system-level electric drivetrain simulation with respect to heat exchange between components', *19th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), IEEE.*

Dankwa, S. and Zheng, W. (2019) 'Twin-delayed DDPG: A deep reinforcement learning technique to model a continuous movement of an intelligent robot agent', *Proceedings of the 3rd international conference on vision, image and signal processing*.

Filizadeh, S. (2013) *Electric machines and drives: principles, control, modeling, and simulation*. CRC press.

Gedlu, E. G., Wallscheid, O. and Böcker, J. (2020) 'Permanent magnet synchronous machine temperature estimation using low-order lumped-parameter thermal network with extended iron loss model', *The 10th International Conference on Power Electronics, Machines and Drives (PEMD 2020). IET.*

Guemo, G. G., Chantrenne, P. and Jac, J. (2013) 'Parameter identification of a lumped parameter thermal model for a permanent magnet synchronous machine', *2013 International Electric Machines & Drives Conference*. pp. 1316-132.

IEC/TS 60349-3 (2010) 'Electric traction - Rotating electrical machines for rail and road vehicles - Part 3: Determination of the total losses of converter-fed alternating current motors by summation of the component losses', https://webstore.iec.ch/publication/1830

Huber, T., W. Peters and J. Böcker (2014) 'Monitoring critical temperatures in permanent magnet synchronous motors using low-order thermal models', *2014 International Power Electronics Conference (IPEC-Hiroshima 2014-ECCE ASIA)*, IEEE.

Kirchgässner, W., Wallscheid, O. and Böcker, J. (2019) 'Deep residual convolutional and recurrent neural networks for temperature estimation in permanent magnet synchronous motors', *2019 IEEE International Electric Machines & Drives Conference (IEMDC).* IEEE.

Kirchgässner, W., Wallscheid, O. and Böcker, J. (2021) 'Data-driven permanent magnet temperature estimation in synchronous motors with supervised machine learning: A benchmark', *IEEE Transactions on Energy Conversion*, 36(3), pp. 2059-2067.

Kirchgässner, W., Wallscheid, O. and Böcker, J. (2023) 'Thermal neural networks: lumped-parameter thermal modeling with state-space machine learning', *Engineering Applications of Artificial Intelligence*, 117, p. 105537.

Kral, C., Haumer, A. and Lee, S. B. (2013) 'A practical thermal model for the estimation of permanent magnet and stator winding temperatures', *IEEE Transactions on Power Electronics*, 29(1), pp. 455-464.

Maroteaux, A. (2016) *Study of analytical models for harmonic losses calculations in traction induction motors*. KTH, School of Electrical Engineering (EES).

Nasir, B. A. (2020) 'An Accurate Iron Core Loss Model in Equivalent Circuit of Induction Machines', *Journal of Energy*, 2020, pp. 1-10.

Nategh, S., A. Boglietti, Y. Liu, D. Barber, R. Brammer, D. Lindberg and O. Aglen (2020) 'A review on different aspects of traction motor design for railway applications', *IEEE Transactions on Industry Applications,* 56(3): 2148-2157.

Nicola, M. and Nicola, C.-I. (2021) 'Improvement of PMSM control using reinforcement learning deep deterministic policy gradient agent', *21st International Symposium on Power Electronics (Ee)*, pp. 1-6.

Ramakrishnan, R., Islam, R., Islam, M. and Sebastian, T. (2009) 'Real time estimation of parameters for controlling and monitoring permanent magnet synchronous motors', *2009 IEEE International Electric Machines and Drives Conference*. pp. 1194-1199.

Sciascera, C., Giangrande, P., Papini, L., Gerada, C. and Galea, M. (2017) 'Analytical thermal model for fast stator winding temperature prediction', *IEEE Transactions on Industrial Electronics*, 64(8), pp. 6116-6126.

Sutton, R. S. & Barto, A. G. (2018) *Reinforcement learning: An introduction*. MIT press.

Wallscheid, O. and J. Böcker (2015) 'Design and identification of a lumped-parameter thermal network for permanent magnet synchronous motors based on heat transfer theory and particle swarm optimisation', *17th European Conference on Power Electronics and Applications (EPE'15 ECCE-Europe)*, IEEE.

Wallscheid, O. (2021) 'Thermal monitoring of electric motors: State-of-the-art review and future challenges', *IEEE Open Journal of Industry Applications*, 2, pp. 204-223.

Wilson, S. D., Stewart, P. and Taylor, B. P. (2010) 'Methods of resistance estimation in permanent magnet synchronous motors for real-time thermal management', *IEEE Transactions on Energy Conversion*, 25(3), pp. 698-707.

Wlas, M., Krzeminski, Z. and Toliyat, H. A. (2008) 'Neural-network-based parameter estimations of induction motors', *IEEE Transactions on Industrial Electronics*, 55(4), pp. 1783-1794.

Xiao, S. and Griffo, A. (2020) 'Online thermal parameter identification for permanent magnet synchronous machines', *IET Electric Power Applications*, 14(12), pp. 2340-2347.

Zhu, Y., Xiao, M., Lu, K., Wu, Z. and Tao, B. (2019) 'A simplified thermal model and online temperature estimation method of permanent magnet synchronous motors', *Applied Sciences*, 9(15), p. 3158.